Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1.-6. (Canceled)

7. (Currently Amended) A method in a computer system for providing an interface between at least one logical view of a relational database and a plurality of normalized relational database tables having stored data, comprising:

receiving an indication of an operation to be performed on the at least one logical view, the at least one logical view including at least one combination of tables from a plurality of normalized relational database tables, the combination previously constructed without reference to foreign keys in the plurality of tables;

automatically determining ~~a~~the corresponding plurality of normalized relational database tables that were used to construct the at least one logical view; and

transparently and automatically causing the stored data in the corresponding plurality of normalized relational database tables to be adjusted in accordance with the indicated operation, thereby automatically performing the indicated operation on underlying data that corresponds to the logical view.

8. (Previously Presented) The method of claim 7 wherein the automatically determining the corresponding plurality of normalized relational database tables comprises:

retrieving from a definitions file a plurality of stored tables and attributes that correspond to a join operation used to construct the at least one logical view.

9. (Previously Presented) The method of claim 7 wherein the transparently and automatically causing the stored data in the corresponding plurality of normalized relational database tables to be adjusted in accordance with the indicated operation, further comprises:

retrieving, from a definitions file associated with the at least one logical view, computer instructions for performing a database insert operation on at least one of the corresponding plurality of normalized relational database tables in accordance with the indicated operation; and

automatically performing the computer instructions to adjust the underlying data in accordance with the indicated operation.

10. (Previously Presented) The method of claim 7 wherein the indicated operation is at least one of a create object or a delete object operation.

11. (Previously Presented) The method of claim 7, further comprising:

receiving an indication of an event from one of the normalized relational database tables;

determining at least one corresponding logical view that was constructed using the one of the normalized tables; and

automatically notifying the determined logical view of the indicated event.

12. (Previously Presented) The method of claim 7, further comprising:

receiving an indication of an expression that includes at least one of a mathematical expression, a function, an attribute reference, or a conditional operation; and

performing the expression to generate at least one of a retrieved data from the database, a value, or a virtual attribute in the at least one logical view.

13. (Previously Presented) The method of claim 7 wherein the transparently and automatically causing the stored data in the corresponding plurality of

normalized relational database tables to be adjusted in accordance with the indicated operation, further comprises:

using a join cache to persistently maintain linkages between the stored tables and attributes and the at least one logical view.

14. (Previously Presented) The method of claim 13, further comprising:

determining whether the data in columns represented by the join cache has changed; and

when it is determined that the data in the columns represented by the join cached has change, recalculating the join thereby updating the at least one logical view.

15. (Currently Amended) A computer readable memory medium containing content that enables a software application in a computing device to manipulate relational database data through logical views, by performing:

receiving an indication of an operation to be performed on the at least one logical view, the at least one logical view including at least one combination of tables from a plurality of normalized relational database tables, the combination previously constructed without reference to foreign keys in the plurality of tables;

automatically determining ~~a~~ the corresponding plurality of normalized relational database tables that were used to construct the at least one logical view; and

transparently and automatically causing the stored data in the corresponding plurality of normalized relational database tables to be adjusted in accordance with the indicated operation, thereby automatically performing the indicated operation on underlying data that corresponds to the logical view.

16. (Previously Presented) The computer-readable medium of claim 15 wherein the computer-readable medium is at least one of a memory of the computing device or a data transmission medium transmitting to the computing device a generated data signal containing the contents.

17. (Previously Presented) The computer-readable medium of claim 15 wherein the contents are instructions that, when executed, cause the computing device to perform the method.

18. (Currently Amended) A software tool for enabling a software application in a computing device to manipulate relational database data through logical views of a relational database, comprising:

a definitions file having a plurality of sections that define a correspondence between stored data in a plurality of stored normalized relational database tables and one or more logical view tables, such that at least one of the logical view tables is constructed without using a foreign key reference from any of the plurality of tables; and

a database join engine configured to automatically interface between at least one logical view table and a corresponding plurality of stored normalized tables of the relational database such that, when an operation is performed on the at least one logical view table, underlying stored data is automatically manipulated in the corresponding plurality of stored normalized database tables.

19. (Previously Presented) The software tool of claim 18, wherein the definitions file further comprises a join cache definition that indicates a cache to maintain persistent linkages between attributes in the stored normalized tables and the at least one logical view.

20. (Previously Presented) The software tool of claim 18 wherein the operation is at least one of a create object or a delete object operation.

21. (Currently Amended) A computer-implemented method for generating an object of any class according to an abstract object model for use as a reusable software component, comprising:

instantiating code for the object that corresponds to a plurality of uniform functions;

instantiating code for the object that corresponds to at least one abstraction conversion tool, the abstraction conversion tool driven by a definitions file having a plurality of sections that define a correspondence between stored data and one or more objects representing at least some of the stored data such that the definitions file may be changed without recompiling code for the one or more objects; and

instantiating code for the object that implements a plurality of object behaviors including notification of a change to an attribute of the object, notification of an addition of an attribute to the object, and notification of a removal of an attribute of the object.

22. (Previously Presented) The method of claim 21 wherein the instantiating code that corresponds to the plurality of uniform functions comprises:

instantiating code for the object that implements functions to read an attribute of the object, write an attribute of the object, get attribute names of the object, invoke a behavior on the object, and get related objects.

23. (Previously Presented) The method of claim 21, further comprising:

instantiating code for the object that implements a plurality of object behaviors including notification of adding an object to a set of objects and notification of deleting an object from a set of objects.

24. (Previously Presented) The method of claim 21 wherein the instantiating code for the object that corresponds to at least one abstraction conversion tool further comprises:

instantiating code for the object the corresponds to an abstraction conversion tool that implements operations that correspond to at least one of object masquerading, object selection, object projection, object concatenation, object joins, property calculations, object bridging, object monitors, or system wrappers.

25. (Previously Presented) The method of claim 21 wherein the generated object is a join engine that implements logical views of normalized tables in a relational database.

26. (Previously Presented) The method of claim 21 wherein the generated object is used to create a powerflow application.

27. (New) The method of claim 7 wherein the stored data has multiple records, each record having multiple attributes, and wherein the at least one logical view includes at least one of a virtual attribute that is not stored in the plurality of database tables, an enumeration of attribute values that convey a logical meaning outside of the database tables, or one or more restrictions to limit which records in a table from the plurality of database tables are included in the logical view.

28. (New) The method of claim 21 wherein the stored data comprises data stored in a plurality of relational database tables.